



TITLE:

# Object-Oriented Approach to Discourse Understanding

AUTHOR(S):

Numaoka, Chisato; Nagano, Tadashi; Tokoro, Mario

---

CITATION:

Numaoka, Chisato ...[et al]. Object-Oriented Approach to Discourse Understanding. 数理解析研究所講究録 1989, 709: 33-47

ISSUE DATE:

1989-12

URL:

<http://hdl.handle.net/2433/101662>

RIGHT:

# Object-Oriented Approach to Discourse Understanding

Chisato Numaoka, Tadashi Nagano and Mario Tokoro

Department of Electrical Engineering  
Keio University  
3-14-1 Hiyoshi, Yokohama 223, Japan

## Abstract

In this paper, We explain a discourse understanding model, ADAM (Actual Discourse Assistant Model) [Numaoka 1988] and show the example of its practical use. In the computer system which has an knowledge base, it is a very important issue how a user communicates with the system. Natural language is one of the way to communicate with it, and the discourse understanding is necessary to comprehend natural language deeply.

ADAM gives a more intuitive expression of the real world by introducing the object-oriented concept into situation semantics. In this model, an *object* has about the same meaning as the one which is called a *parameter* in situation semantics. An object is a unit of information to be operated by a knowledge system. All the information, such as the character of individual or a causal relation among objects, are extracted from the objects in a knowledge base. Thus, in ADAM, a knowledge system which has a natural language user interface system can cope with lots of user's requests by utilizing these information.

## 1 Introduction

Discourse understanding is an indispensable field in understanding natural language. Situation semantics [Barwise 1983] is a very attractive theory to explain some of the problems in discourse understanding by using situation and set theories. One of the features of situation semantics is that described situations are interpreted within a relationship between real situations and abstract situation types. Owing to this feature, situation semantics can explain the reason why the difference of situation causes the difference of interpretation.

Actually, many studies in discourse understanding are being carried out by using this feature of situation semantics. DUALS [Yasukawa et al. 1985] and SCORE

[Nagase 1986] are examples of discourse understanding systems which use situation semantics and utilize logic programming. The situated language research program in CSLI [CSLI 1987] is an integrated study based on situation semantics. In this program, many fundamental and applied studies for situation semantics are being carried out.

Without utilizing natural language, various knowledge systems such as expert systems have been produced, and applied for practical use. We would like to have a natural language interface for such knowledge systems. Natural language gives more friendly and flexible man-machine interface than existing command languages. The problem to utilize natural language in knowledge system is its ambiguity in discourse. One of the ambiguity results in personality of human-being. Situation semantics is valid for capturing this kind of ambiguity in discourse.

This present paper takes a pragmatical approach to representing the framework for discourse understanding. That is to say, we propose a method which will be practically used for the framework of situation semantics to be applied to a natural language user interface for a knowledge system.

ADAM (Actual Discourse Assistant Model) is an extended model of situation semantics in order to build a natural language interface system based on the idea of object oriented knowledge based system. Extension to situation semantics is made in representing an object in discourse by using knowledge objects, and extracting information from the relation between objects and from object itself. In order to understand a discourse, we construct a model of a speaker's mind within an individual discourse situation and to utilize the user model in order to understand the utterances of him.

In this paper, Section 2 presents an overview of ADAM. Section 3 gives a example of natural language man-machine interface for the behavior simulation system.

## 2 ADAM: A Model of Discourse Understanding

ADAM is not equal to situation semantics but is based on it. We now added more object-oriented interpretations to situation semantics. Situation semantics is a very effective theory to account for differences between interpretations by using differences between situations. As we have already stated in the introduction, however, we do not think that situation semantics gives an adequate framework for manipulating objects in discourse. Thus, we would like to construct a more intuitive model than situation semantics for the actual object in the real world using the object-oriented approach. In this section, we will present ADAM for discourse understanding.

## 2.1 World View of ADAM

Before we will define the specifications of ADAM, we have to state the following assumption:

**(Assumption 1)** The information conveyed in a discourse mentions the objects in the discourse.

Based on this assumption, we give the following three definitions:

**(Definition 1-1)** For some object  $i$ , if

$$i \in \text{dom}(f)$$

∨

$$i \notin \text{dom}(f) \wedge i \in \text{dom}(f') \wedge f \longrightarrow f'$$

then

object  $i$  can be referred to from the viewpoint of domain  $f$ ,

where

both  $f$  and  $f'$  are discourse situations.

Note:  $f \longrightarrow f'$  means that  $f$  can refer to  $f'$ .

**(Definition 1-2)** An object is either of the following resources:

primitive objects or situation.

## 2.2 Primitive Objects

In ADAM, the followings are primitives:

- Temporal-Location (TL): The location on the time-axis.
- Spatial-Location (SL): The location in the 3-dimensional space.
- Individual (Ind): Singular NP, i.e., entity.

These primitives are defined as objects in object-oriented programming. For example, the following sentence,

Bob is playing with John.

is represented as Figure 1.

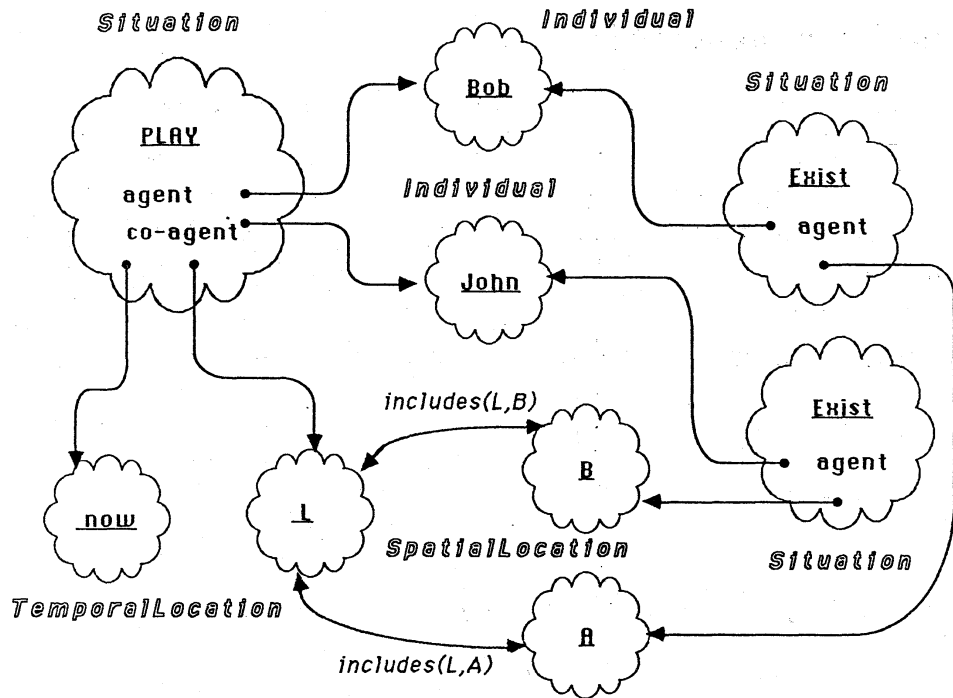


Figure 1: The object-oriented description for a situation

### 2.2.1 Temporal-Location and Spatial-Location

In ADAM, two types of representation of time are given. One is the absolute temporal-location, the other is the relative temporal-location.

- Absolute Temporal-Location: a time is *absolute* if it is specified. For example,

*I must be in Tokyo before 10:00 a.m.*

Current (computer-)system time is also absolute temporal-location. Therefore, the time when the utterances are made is of this type.

- Relative Temporal-Location: a time is *relative* if it is not specified in a sentence. For example,

*She said that she was beautiful.*

In this sentence, the speaker did not indicate when she was so. However, some time has to be included in the situation described in the above sentence. Therefore, time  $t$  is particularly prepared to be assigned to that situation. It is given the constraint that

*whollyPreceeds(t, now),*

where 'now' indicates the current time. We call this time *relative temporal-location*.

The temporal-location which we have defined here is different from that of interval temporal logic [Moszkowski 1983]. That is, in ADAM, time is not regarded as divided fields but points on a continuous axis.

There are also two types of location in Spatial-Location. One is Absolute Spatial-Location, the other is Relative Spatial-Location.

- Absolute Spatial-Location: a point in a 3-dimensional space is *absolute* if it is specified. For example,

*He was born in Kyoto.*

- Relative Spatial-Location: a point in 3-dimensional space is *relative* if it is not specified in a sentence. For example,

*The player was pleased because his shot was successful.*

In this case, if some place has already been mentioned before, that place is used for this situation. Otherwise, an adequate spatial-location will be assigned to the situation.

### 2.2.2 Individual

An individual is either a physical or a conceptual entity. Each individual in a discourse situation is a particular existence in the real world. That is, each individual in the world is unique. For example, in some discourse situation, there is an instance of a human-being named John. There is also an instance of dog name John. Both of their names are John, but they are different individuals. Besides, in ADAM, an individual has features. Thus, if we can access an object, we can obtain the features of the object.

## 2.3 Discourse Situation

In situation semantics, a discourse situation is explained to be a specific situation which includes two specific roles, the speaker and the addressee, and described situations. We agree with it on this point. Situation semantics, however, does not give more details about the discourse situation. For instance, it does not give an explanation of the hierarchical relation of discourse. The hierarchical relation of discourse is shown in the following example:

*John said to Bob, "Mary says that she loves you."*

In this case, the discourse situation:

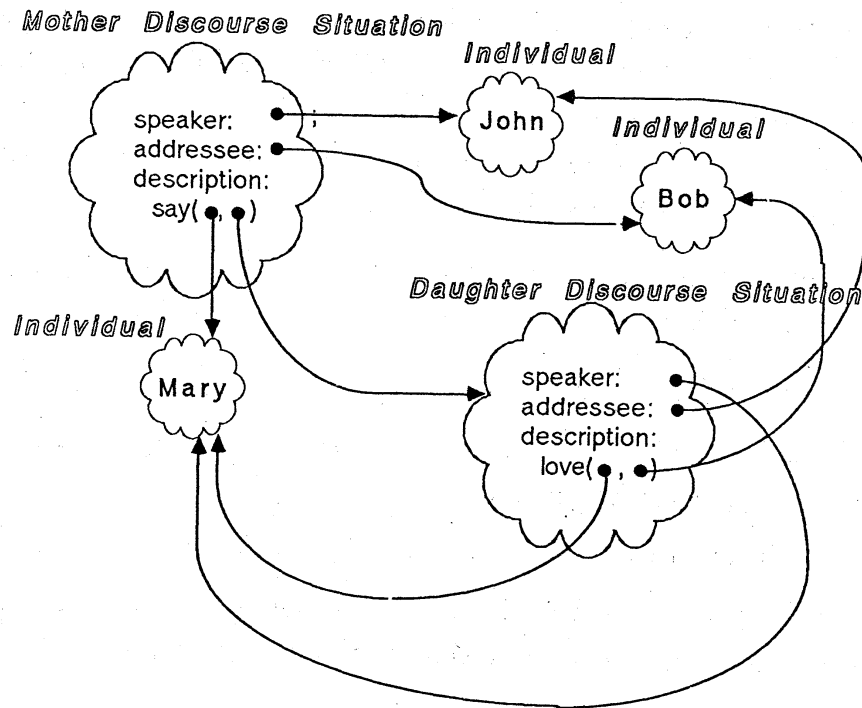


Figure 2: The hierarchical relation in discourse situation

DSIT := at l: speaking, Mary; yes  
 addressing, John; yes  
 saying, Mary, "I love Bob"; yes

is included in the discourse situation:

DSIT := at l: speaking, John; yes  
 addressing, Bob; yes  
 saying, John, "Mary says that she loves you"; yes

These are shown in Figure 2.

In ADAM, this kind of expression is defined by the following description:

**(Definition 2-1)** A discourse situation is treated as an event. Therefore, the situation is a transitional one. This causes a series of sentences in the discourse situation to be described in the frame of the situation.

**(Definition 2-2)** The top level discourse situation (the situation between computer the system and its user, e.g.) has the following frame:

speaker: user

addressee: computer-system  
temporal-location: now  
spatial-location: here  
described-situation: ...

(Definition 2-3) If a discourse situation exists in some other discourse situation as a described situation, the former is called *daughter discourse situation* and the latter is called *mother discourse situation*. Hence, a mother discourse situation can refer to the daughter discourse situations while a daughter discourse situation cannot refer to the mother situations.

## 2.4 Assumptive Situation

In discourse, a sentence may give an attitude report. This type of sentence does not describe a real situation. For example,

*I believe that he will pass the examination.*

In this sentence, the fact that the speaker believes that someone will pass the examination is described. However, the situation in which the speaker believes may not be realized. It is only an assumption. Accordingly, this type of situation should be expressed as a situation object, this is because a situation object is made as the result of a situation being realized. In the utterance,

*Though I wanted to go to the States, I did not go there after all.*

the description "*I wanted to go to the States*" includes the description "*I go to the States*". This embedded description is expressed as follows:

*situation(go, I, the States; 1).* (1)

Here, though it is assumed that this situation occurs at some temporal-location TL in the past, TL is not fixed yet. The remainder of the above utterance includes the description "*I did not go*". This is expressed as follows:

*situation(go, I, the States; 0).* (2)

Here, though it is also assumed that the situation occurs at some temporal-location TL' in the past, TL' is not fixed yet, either. If we assume that the situation objects (1) and (2) are referring to the same real situation, a contradiction occurs. Accordingly, situation objects (1) and (2) are not the same (i.e., multiple situation objects, which are referring to the same real situation, cannot exist). However, If the situation object (1) and (2) are different objects, the real situations which correspond to these objects



are also different. This indicates that the two different situations occur simultaneously. These two situation objects have the same individuals, the same temporal location, and the same spatial location. The difference between these objects is only a truth-value. This is impossible in the real world, if we do not assume multiple world. Thus, this is again a contradiction.

In order to avoid the contradiction, we introduced the idea of the assumptive situation in ADAM. This type of situation object represents the situation whose occurrence is not realized yet. The assumptive situation is described as follows:

**(Definition 3-1)** An assumptive situation object describes a situation other than that which has already occurred or is occurring now (i.e., an interrogative sentence).

**(Definition 3-2)** A situation object has priority over an assumptive situation object. That is, if an assumptive situation object *A* and a situation object *S*, which is unifiable with the assumptive situation except for the truth-value, exist, we can assume that they describe the same real situation.

**(Definition 3-3)** In addition to Definition 3-2, if the truth-values are the same, the situation object in which the assumptive situation object is included is called a *successful situation*.

An assumptive situation can be also used as a daemon which operates on an expected in the future such as the following sentence indicates:

*If the temperature of this room is higher than 20 degree centigrade, switch on the cooler.*

### 3 Example: Natural Language Interface for Behavior Simulation System

In this section, as an example, we show how to make a natural language interface based on ADAM for a behavior simulation system, PARADISE [Maruichi 1987]. We use LISP like expression form in order to show one of the implementation of ADAM. In this form, the expression:

$(\leftarrow \text{object message})$

indicates that an interface system sends a *message* to an *object*. Behavior simulation simulates the situational transition produced by the interactive actions between participants in the simulation. Here, the participants are also objects. For example, two types

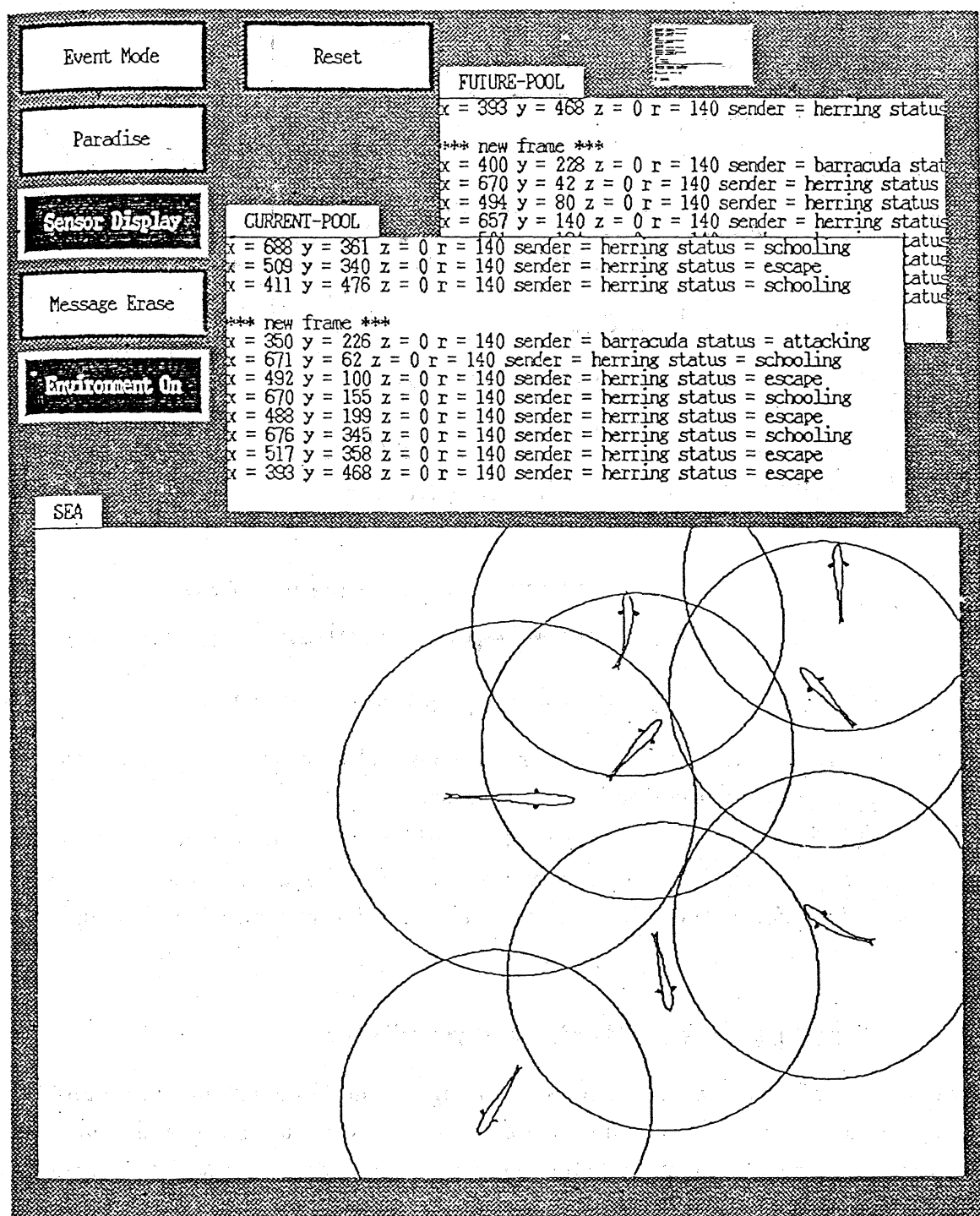


Figure 3: An example of a behavior simulation

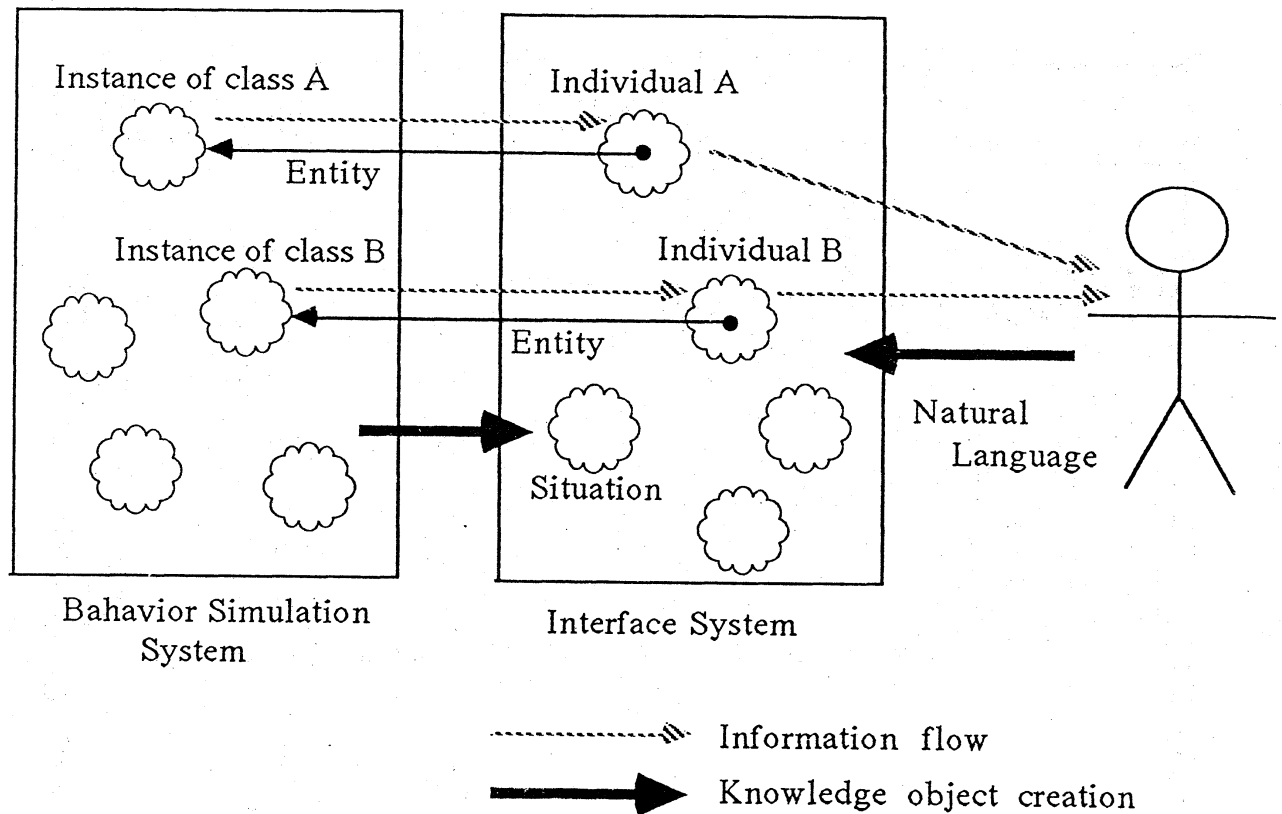


Figure 4: Interfacing behavior simulation

of fish, a herring and a barracuda, in Figure 3 are objects as real world entities, which have knowledge about their actions and status.

The interface system is to the simulated world what we, human-being, are to the real outer world. That is, this simulated world is an operational one for a natural language interface. Therefore, we can ask the system to do a job instead of us. This is shown in Figure 4.

### 3.1 Description of Interface Knowledge

We have to differentiate an object as knowledge in the discourse understanding (e.g. in the interface system) from simulated objects. Let us hereafter call a simulated object *an actor*. Asking the interface system to operate an actor, we must define the following items:

- (1) Items related to the objects as knowledge.
- (2) Items related to the concept on the natural language for operations.

Figure 5 shows an example of the definitions for the above items: In this definition, it is defined that the relation between a word in our natural language, such as *create*, and a method of the specific object, such as *:create*. Using this definition, the following expression: the expression:

```

(def-interface
  (topic      behavior simulation")
  (speaker    any)
  (individual "herring"
    (entity   Herring)
    (operation
      (concept
        ("create" :create))
      (instance
        ("position" :position)
        ("go-up" :up)
        ("go-down" :down)
        ...
      )))
  (individual "barracuda"
    (entity   Barracuda)
    (operation
      (concept
        ("create" :create))
      (instance
        ("position" :position)
        ("go-up" :up)
        ("go-down" :down)
        ...
      )))
  ))))

```

Figure 5: Definition of the Interface between the Knowledge and the Actor

*I'd like to create a barracuda.*

will make the following method in that discourse situation:

*(← barracuda :create)*

Assume the following utterances:

- Let's talk about behavior simulation. (1)
- There are three herrings in this display. (2)
- These are this, this, and this. (3)
- (After this utterance, the specific picture of herring  
will be pointed with some pointing device such as mouse.)
- (While pointing at the picture of barracuda.)
- This is a barracuda. (4)

In utterance (1), the interest of the interface system is in the behavior simulation. That is, the topic is established as a behavior simulation. As a result, the definition of the topic *behavior simulation* is loaded into the event manager in the interface system. When sentence (2) is uttered, an individual as an instance of a herring was created in the interface system. Utterance (3) plays a roll to relate the individual to the simulated object.

### 3.2 Meta-programming by Natural Language

One of the efficient uses of natural language is meta-programming <sup>1</sup>. In this system, the assumptive situation object make meta-programming possible.

*Please notify me when any herring will be eaten by the barracuda.*

This utterance makes an assumptive situation object as can be seen in Figure 6, and the existence of the assumptive situation object lets the interface system observe the operation on the behavior simulation system.

### 3.3 Dynamic Change of the Factors in the Simulation System

In a simulation system, we often want to change at will a factor in the simulation. This especially applies to behavior simulation because factors should be visually determined with respect to the current state of simulation stage setting. In this system, a flexible setting of the factors is possible by using natural language. For example, the expression:

---

<sup>1</sup>Here, meta-programming means something like reflective programming [Maes 1987]. That is, the behavior of an object can be controlled by some meta-object, while object itself is not changed.

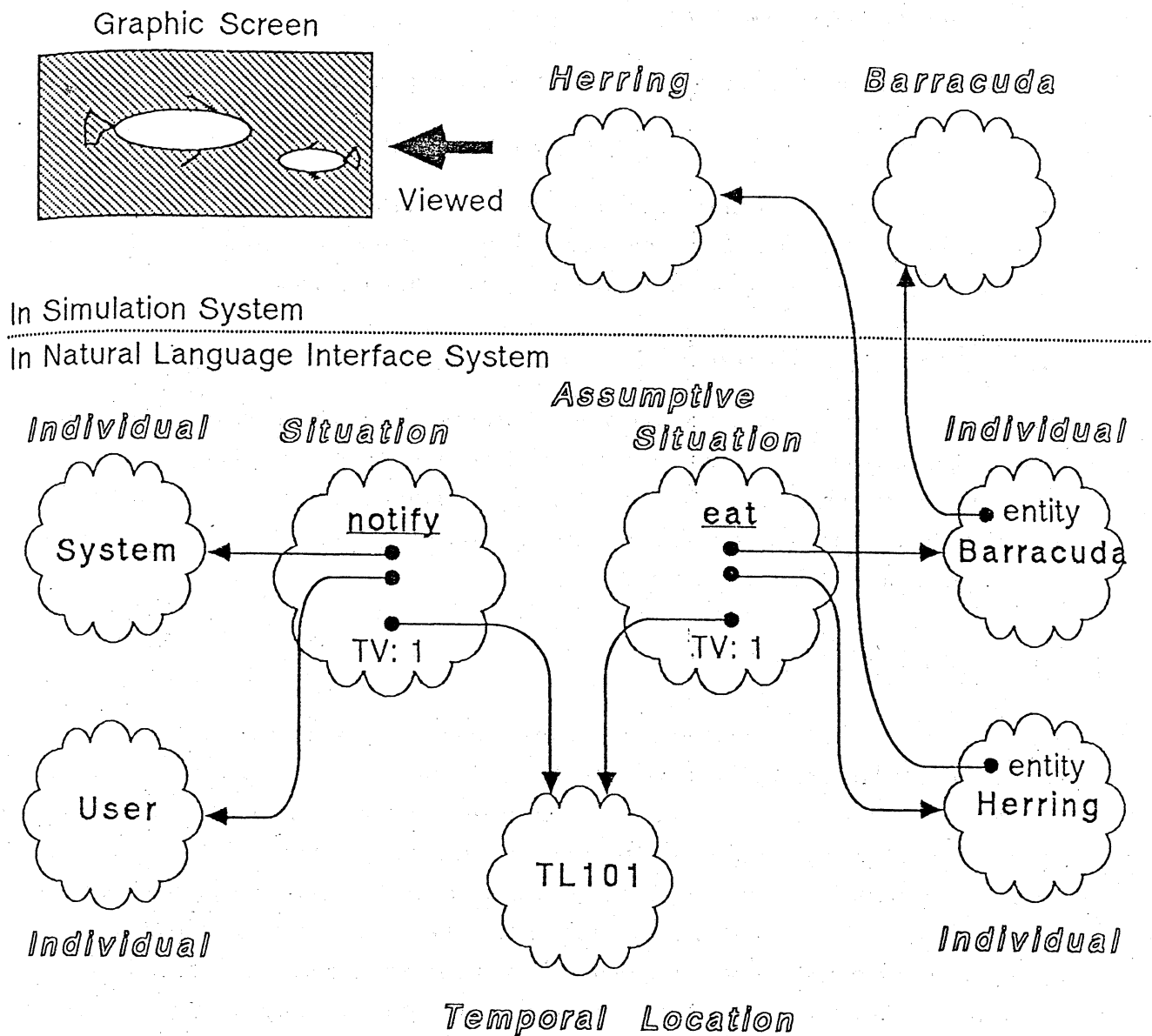


Figure 6: Creation of an assumptive situation object as a meta-programming: The situation object, "the system notify the user," is evaluated, if its embedded assumptive situation is realized (i.e., if the event that the barracuda eats a herring occurs in the behavior simulation system).

*Move the herring up a little.*

may have the same effect as the following message expression:

$(\leftarrow \text{herring} : \text{move} (+ (\leftarrow \text{herring} : \text{position}) (10\ 10)))$

However, the former is a more flexible expression than the latter. This is because, in the former, the same action can be repeated by an utterance such as "*a little*". This kind of expression is one of the merits of using natural language.

### 3.4 Making a Personal Environment

In ADAM, the discourse situation object corresponds to the speaker. Therefore, a discourse situation object for one speaker differs from that for another. As knowledge which is referenced from an object is utilized for discourse understanding, a difference between the objects causes a difference in the interpretation of utterances. This means that a personal interface can be built.

## 4 Conclusion

In this paper, we have explained our discourse understanding model, ADAM. Our motivation for developing ADAM is to clarify the significance of objects such as an individual, a space-time location, and so forth, in discourse understanding and to build a model in order to make a natural language user interface system. Since objects in situation semantics are treated as parameters, the characters of these objects themselves cannot be illustrated. We are rather interested in the characteristics of an object itself. In knowledge based system, it is important how information relate with a real situation conveyed by natural language are expressed. Thus, our first objective is to represent the real world using objects, and to interpret the relation between objects.

We believe we have achieved the objective to some extent as the above example show. By using discourse situation, ADAM supports the user specific environment in which the knowledge based system understands information which are taken from the relation among objects or a object itself. Additionally, by using assumptive situation, we can add a daemon which executes specific operation in a knowledge system in the future. Currently, using ADAM, we are implementing a discourse understanding system called ASUKA [Numaoka 1987].

## References

[Barwise 1983] Barwise, J., J. Perry, *Situation and Attitudes*, MIT Press, 1983.

- [CSLI 1987] CSLI, *Fourth Year Report of the Situated Language Research Program*, Report No.CSLI-87-111, 1987.
- [Maes 1987] Maes, P., *COMPUTATIONAL REFLECTION*, PhD thesis, Vrije Universiteit Brussel, AI-LAB Pleinlaan 2, B-1050 Brussels, Belgium, Mars 1987.
- [Maruichi 1987] Maruichi, T., Uchiki, T., and Tokoro, M., *Behavioral Simulation Based On Knowledge Objects*, Conference Proceedings of ECOOP-87, 1987.
- [Moszkowski 1983] Moszkowski, B., Z. Manna, *Reasoning in Interval Temporal Logic*, Stanford University, Department of Computer Science, Report No.STAN-CS-83-969, 1983.
- [Nagase 1986] Nagase, T., H. Mizoguchi, *An Attempt to Design Discourse Understanding System Based on Situation Theory*, Proc. of the Third Annual Conference of JSSST, pp.37-40(in Japanese), 1986.
- [Numaoka 1987] Numaoka, C., M. Tokoro, *ASUKA: Feature and Specification of Discourse Understanding System*, Proc. of the fourth annual conference of JSSST, pp.55-58 (in Japanese), 1987.
- [Numaoka 1988] Numaoka, C., M. Tokoro, *ADAM: An Extension of Situation Semantics for Practical Use*, to appear in FGCS'88, 1988.
- [Yasukawa et al. 1985] Yasukawa, H., Hirakawa, H., Mukai, K., Miyoshi, H. and Tanaka, Y., *The Outline of Discourse Understanding System DUALS*, Tech.Memo, TM-0118, 1985.